Real-time Identification of Respiratory Movements through a Microphone

Juan Castro[a], Pere Marti-Puig[b]
[a] Universitat Oberta de Catalunya
[b] University of Vic-Central University of Catalonia

KEYWORD

Digital signal processing
Pattern recognition
Machine learning
Respiratory analysis
Acoustic features

ABSTRACT

This work presents a software application to identify, in real time, the respiratory movements -inspiration and expiration- through a microphone. The application, which has been developed in Matlab and named ASBSLAB for the GUI version and ASBSLABCONSOLE for the command-line version, is the result of a research and experimentation process. A total of 48 minutes of breathing movements from four subjects was recorded and 18 acoustic features were extracted to generate the data model. A first level of identification, based on the classification of tiny audio segments, was designed using kNN supervised method. The second level of identification implements a state machine that takes the results ordered in the time from kNN as input and identifies the whole respiratory movement, achieving a level of positive identifications above 95%. As computation time is a handicap, the application let the user choose easily the sample rate, the audio segment size and the set of acoustic features to use in the identification process. In addition, based on the number of features selected, this works suggests those that achieve best results.

1 Introduction

Breathing is the physiological process by which organisms capture oxygen and get rid of carbon dioxide. In humans, the respiratory system consists of the respiratory muscles, the lungs and the airways. The exchange of air between the atmosphere and the lungs (inspiration) and vice-versa (expiration) generates an acoustic signal.

Over the course of time, breath sound auscultation informs the specialist about the health of the respiratory system and about possible alterations or diseases. The breakthrough of digital signal processing led to new study mechanisms of the respiratory system, with additional advantages such as continual monitoring, objective diagnostics, remote diagnostics and others.

Signal processing application in respiratory analysis is not recent. Already in 1980's, works from Chowdhury and Majumder [Chowdhury, S.K. & Majumder, A.K. 1981] and Charbonneau et al. [Charbonneau, G. et al., 1982] proposed signal processing techniques in the frequency domain to be applied in the analysis and classification of breath sounds. More recent works have highlighted a variety of approaches to analyse breath sounds for the detection of dysfunctions or diseases as apnoea [PETSATODIS, T. et al., 2012; MOEDOMO, R.L. et al., 2012], asthma [Oud, M. & Dooijes, E.H., 1996; Moedomo, R.L. et al., 2012], snores [Petsatodis, T. et al., 2012; Snider, B.R. & Kain, A., 2013; Yadollahi, A. & Moussavi, Z., 2010], wheezes [Snider, B.R. & Kain, A., 2013; Patel, U., 2011; Moedomo, R.L. et al., 2012] and crackles [Avalur, D.S., 2013; Patel, U., 2011].

Methods belonging to Artificial Intelligence disciplines are used to identify respiratory events. K-Nearest Neighbour [Oud, M. & Dooijes, E.H., 1996], Naive Bayes [Alshaer, H. et al., 2014], Support Vector Machines [Alshaer, H. et al., 2014], Random Forest [Alshaer, H. et al., 2014], Neural Networks

[Yuan, K. et al., 2011; Patel, U., 2011; Mason, L., 2002] and Hidden Markov Models [Snider, B.R. & Kain, A., 2013] are common techniques that have been proven to achieve good results.

A major challenge related to breath sound analysis is its detection. Normal breathing generates a very quiet sound, which is difficult to detect due to ambient noise interference. On the one hand, invasive capture methods like a tracheal microphone [Yadollahi, A. & Moussavi, Z., 2010] or an acoustic sensor within a mask [Avalur, D.S., 2013] are better options to obtain more robust signal but may cause discomfort to the user. On the other hand, non-invasive methods like a microphone oriented toward the user's head at a short distance [Petsatodis, T. et al., 2012; Snider, B.R. & Kain, A., 2013] are easier to build but are more sensitive to external noises. In addition, the use of multiple microphones can be used to focus on breath signal, reducing ambient noise, but may complicate the system for a home based solution.

The focus of this work differs substantially in several aspects from previous ones. Firstly, this work proposes a software solution where users are allowed to configure a set of parameters used for breath sound analysis. Secondly, the solution proposed is open source, enabling developers to customise or to add new features easily. Finally, preceding works treat breath sound analysis and identification as a two steps process, where respiratory sounds are analysed after the whole sound session is recorded. Here, breath sound analysis and identification is made in real time. More precisely, this work develops a software solution that captures the breath sound from a microphone located near the user and detects and identifies the respiratory movements of inspiration and expiration –either mouth or nasal- in real time.

With this approach, it is plausible to think of new non-invasive breathing monitoring systems in practical applications such as:

- Stress measurement in high demanding concentration exercises in cabin simulators.
- The control that a user with asthma is continually breathing with the nose, avoiding mouth breathing.
- Respiratory movement pattern monitoring in relaxing techniques, alerting the user when his respiratory sequence does not match the initial configured pattern.

To achieve the goal of this work, the result of breathing analysis from four subjects has been employed to generate a repository with eighteen acoustic features. Then, proven kNN method and an implemented upper level state machine have been applied to, eventually, identify respiratory movements. The fact that the identification is done in real time, determines the methodology as well as computation time thresholds.

2 Material and Methods

2.1. Breath sound recording and labelling

Breath sounds were recorded from four healthy subjects (two male and two female) in 10 minutes sessions, giving a total of 48 minutes of respiratory movements that corresponded to near 1,800 respiratory events. This would constitute the source for the training data set. Each session included inspirations and expirations, both mouth and nasal. The scenario for the recording was a standard room with the door closed to avoid ambient sound interference from outside. Inside the room, an armchair was provided for the subject. Breath sounds were captured with a cardioid pattern capacitor microphone with 32 dB of sensitivity, mounted on a tripod and pointing towards the face of the subject at a distance of 22 cm approximately. The recording was made at a sampling rate of 44.1 KHz and 16 bits of resolution.

An audio editor was used, from the signal plot in the time domain and its spectrogram, to manually label each respiratory movement. It was detected a continuous noise interference below 100 Hz. This noise prevented, in some subject recordings, from marking breath sounds transitions with enough accuracy. To avoid this, a 6th order elliptic filter with 0.1 dB of ripple and 50 dB of stopband attenuation was applied to remove this noise. The filter would also be applied in the process of respiratory movement identification in real time while capturing its sound.

To experiment with different sampling frequencies, breath sounds recorded were resampled to 22,050 Hz, 16,000 Hz, 11,025 Hz and 8,000 Hz. Moreover, the less was the frequency, the less was the amount of data to process.
The labelling of the respiratory events was a manual process, using the audio editor and two columns in an external spreadsheet. The first column included the time mark from the recorded breathing of the transition point between two respiratory events. The second column labelled the event before the time mark, numerically codified (1 = mouth inspiration, 2 = mouth expiration, 3 = nasal inspiration, 4 = nasal expiration).

2.2. Acoustic features extraction

2.2.1. Windowing

Acoustic features were extracted from tiny segments of size W from the recorded breathing using a moving Hamming window (9) with a 50% overlap. A set of different predefined window sizes (see Table 1) was used according to sampling frequencies and optimizing FFT data points.

Sampling Freq. (Hz)	Window Size (ms)	Window points	FFT points
22,050	92	2,028	2,048
	69	1,521	2,048
	46	1,014	1,024
	23	507	512
16,000	128	2,048	2,048
	96	1,536	2,048
	64	1,024	1,024
	32	512	512
11,025	92	1,014	1,024
	69	760	1,024
	46	507	512
	23	253	256
8,000	128	1,024	1,024
	96	768	1,024
	64	512	512
	32	256	256

Table 1. Window sizes based on sampling frequencies.

2.2.1. Acoustic features

From each segment of size W, the following 18 acoustic features were extracted:

<u>Root Mean Square (RMS)</u>: Also known as the quadratic mean, it is a statistical measure of the magnitude of the sound and it is given by:

$$RMS = \sqrt{\frac{1}{N}\sum_{n=1}^{N} x[n]^2} \qquad (1)$$

The RMS is needed to compare its value to the threshold, below which, audio information should be ignored, as it is lower than a predefined value, which was considered enough, of 4 dB audio signal-to-noise ratio (SNR).
From the analysis of the captured audio from the recording scenario, once have been removed the frequencies below 100 Hz, it is shown random residual noise with a magnitude between -65 dB and -62 dB (relative to amplitude 0). Setting -58 dB (0.00125 of amplitude) as the threshold value, would match the desired SNR value of 4 dB.

<u>Spectrum Centroid (SC)</u>: It is the centre of mass or point of balance of the signal spectrum. As it is an indication of the predominant frequency area of the signal, it is associated to the 'brightness' of the sound, and it is defined by:

$$SC = \frac{\sum_{n=1}^{N}(n \cdot X[n])}{\sum_{n=1}^{N} X[n]} \qquad (2)$$

where n is the index of the nth frequency band of the spectrum and $X[n]$ is the magnitude of that band.

<u>Spectrum Flatness (SF)</u>: It is an indicator of the homogeneity of the signal spectrum and it is calculated from the ratio between geometric and arithmetic means, calculated as:

$$SF = \frac{\sqrt[N]{\prod_{n=1}^{N} X[n]}}{\frac{\sum_{n=1}^{N} X[n]}{N}} \quad (3)$$

Zero-crossing rate (ZCR): This is a temporal feature of the signal and tells us about the number of changes in the sign of the signal normalized by its length, given by:

$$ZCR = \frac{1}{N}\sum_{n=1}^{N} |\operatorname{sgn}(x[n]) - sgn(x[n-1])| \quad (4)$$

where sgn is the sign function.

Roll-Off (RO): This is the frequency, below which, an 85% of the spectrum distribution is concentrated. The value is given by:

$$\sum_{n=1}^{RO} X[n] = 0.85 \cdot \sum_{n=1}^{N} X[n] \quad (5)$$

Cumulative Frequency RMS Ratio: It has been detected that at 2 KHz, the average ratio of cumulative energy between expiration and inspiration is at maximum for mouth respiratory events. This work includes this ratio as a feature.

Mel Frequency Cepstral Coefficients (MFCCs): MFCCs are very common and an efficient technique for signal processing and they have been successfully used in speech analysis [YOUNG, S.J., et al. 1999] among others. It is a representation of the short-term power spectrum of a sound, based on a linear cosine transform of a log power spectrum on a non-linear Mel scale of frequency. They are calculated as:

$$MFCC[m] = \sqrt{\frac{2}{N}}\sum_{k=1}^{N} \log(E[k])\cos(m(k-\frac{1}{2})\frac{\pi}{N}), \quad (6)$$

for

$$1 \le m \le M$$

where:

$$E[m] = \sum_{n=1}^{N} |X[n]| \cdot H[m], \quad 1 \le m \le M \quad (7)$$

$$X[n] = FFT(x[n] \cdot w[n]) \quad (8)$$

being $H[m]$ the Mel filter bank of length M and $w[n]$ the Hamming window, given by:

$$w[n] = 0.54 - 0.46\cos(2\pi\frac{n}{N}) \quad (9)$$

As the first coefficient has a high correlation with the energy, only coefficients 2-13 are extracted.

2.3. Generation of the repository

From the breath sounds recorded audio together with the spreadsheet with its labelling information, the data model was generated. It contains, by applying every combination of sampling rate and window size showed in table 1, all the extracted 18 acoustic features explained before. The data model formed the training data set to which the system would query to label new unknown samples.

2.4. The classification method

This work has been based on the supervised method k-Nearest Neighbour (kNN) [Harrington, P., 2012] to classify samples. kNN method was chosen because it works with numeric data, has a high accuracy, is insensitive to outliers and is very easy to reuse in scenarios where the number of implied features changes.
kNN measures the distance between an unknown item and a set of known items (the training set). Those k items to which the distance is minimal are the result of the classification. For this work, k is equal to 1 and the distance is measured by the Euclidean distance, given by:

$$d(x,y) = \sqrt{\sum_{j=1}^{N}(x_j - y_j)^2} \quad (10)$$

where x_j and y_j are the feature j from a total of N features.
One major drawback in calculating distances is that features have different measurement scales. Features with higher magnitude have predominant weight in the distance measurement. One solution is to standardize the training data by the given formula:

$$z_{ij} = \frac{x_{ij} - \mu_j}{\sigma_j} \qquad (11)$$

where z_{ij} is the standardized value of item i for feature j , x_{ij} is the value before standardization and μ_j, σ_j are the mean and standard deviation of feature j from all item in the training set.
Anyway, as it is a real time solution, standardization is not an optimal technique to homogenise the features measures scale because unknown samples do not participate in the calculation of statistical indicators.

2.5. The state machine

The classification method labels tiny segments of a few tens milliseconds length and, as expected, it would not be 100% accurate. A state machine, which is fed by the ordered output sequence from kNN method, is implemented to eventually identify respiratory movements.
For example, for a rhythm of 40 respiratory movements per minute and a window length of 128 ms, there would be about 20 audio segments per respiratory event to classify. If kNN method would be 100% accurate, its output for a sequence of a mouth inspiration – mouth expiration – nasal inspiration – nasal expiration would be something similar to

0011111111111111100000222222222222000033
333333333300044444444444440000

where a '0' means the audio segment below the energy threshold according to (1) and corresponding to respiratory movement transitions.
However, output from kNN could be more probably something like

001221122411111100000221112221220000031
13333444330004333414444440000

due to some errors in the labelling process.
Here in where the state machine takes part. It calculates some basic statistical indicators to eventually decide the respiratory movement detected. Its implementation took into consideration a new time domain feature, based on the phases of rising and decay of energy in respiratory movements.

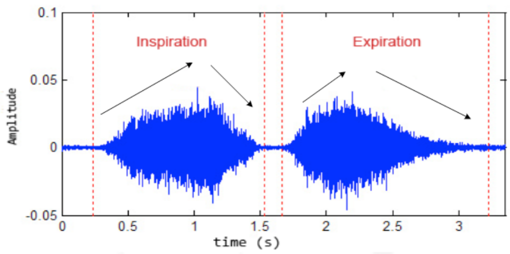

Fig. 1. Rising and decay of inspiration and expiration movements.

As shown in figure 1, rising is normally longer in inspiration than in expiration, whereas the opposite occurs with decay.

Figure 2 shows the state diagram of the implemented state machine.

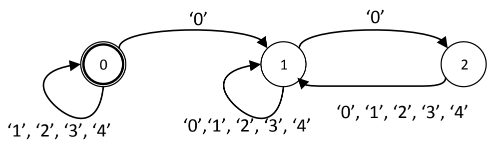

Fig. 2. State machine for respiratory movement identification

The initial state is state 0. It is the beginning of one respiratory movement identification. Identification must start at the initial phase of the respiratory movement, so it remains in state 0 until a segment labelled '0' has arrived. State 1, starts from the moment a non-'0' labelled segment arrives and updates the following internal indicators from the successive output from kNN classification until it receives a new classified '0' segment:

- Consecutive number of non-'0' segments with rising energy.
- Consecutive number of non-'0' segments with decay energy.
- Consecutive number of segments labelled as '1'.
- Consecutive number of segments labelled as '2'.
- Consecutive number of segments labelled as '3'.
- Consecutive number of segments labelled as '4'.
- Consecutive non-'0' labelled segments.
- Time elapsed since the last '0' labelled segment.

State 2 takes the decision of the respiratory movement captured, based on the following calculation:

N = NumOf_1 + NumOf_2 + NumOf_3 + NumOf_4;
p1 = NumOf_1 / N;
p2 = NumOf_2 / N;
p3 = NumOf_3 / N;
p4 = NumOf_4 / N;
pMouth = (NumOf_1 + NumOf_2) / N;
pNasal = (NumOf_3 + NumOf_4) / N;
pIns = (NumOf_1 + NumOf_3) / N;
pExp = (NumOf_2 + NumOf_4) / N;
RisingDecay = NumRising + NumDecay;
pIns2 = NumRising / RisingDecay;
pExp2 = NumDecay / RisingDecay;
pMouthIns = 0.70 * p1 + 0.10 * pMouth + 0.10 * pIns + 0.10 * pIns2;
pMouthExp = 0.70 * p1 + 0.10 * pMouth + 0.10 * pExp + 0. 10 * pExp2;
pNasalIns = 0.70 * p1 + 0.10 * pNasal + 0.10 * pIns + 0.10 * pIns2;
pNasalExp = 0.70 * p1 + 0.10 * pNasal + 0.10 * pExp + 0.10 * pExp2;
Event = max(pMouthIns,pMouthEsp,pNasalIns pNasalEsp);

2.6. Implementation

The solution was implemented in Matlab version 7.12.0.365 (R2011a) using a laptop computer with Intel Core i7 2GHz and 8 GB of RAM.

In order to save computation time, common calculations are executed at the time the sampling rate, the window size and the set of selected acoustic features are chosen. These operations included, among others, intermediate calculations for MFCC and initialization of the kNN through *knncreatens* Matlab function.

3 Results/Discussion

3.1. Acoustic features analysis

Distributions of the four respiratory movements of the the group of selected acoustic features mentioned in the previous section are shown in Figures 3 and 4. As expected, spectrum centroid was higher in nasal breaths than in mouth breaths. Also, the spectrum flatness showed that nasal breath took more bandwidth while mouth breath concentrates its energy in much lower bands.

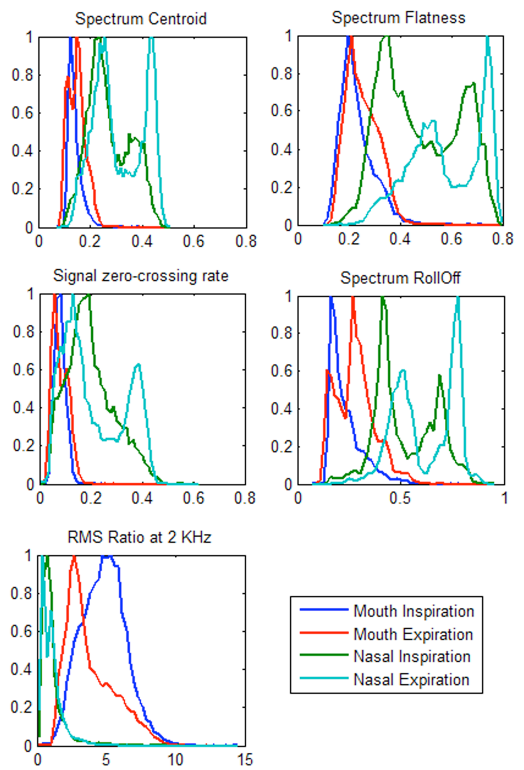

Fig. 3. Plot of extracted acoustic features (except MFCC) showing normalized distributions from the 4 respiratory movements analysed at 22,050 Hz of sampling rate and 92 ms of window size.

However, as individual distributions were clearly differentiated for nasal and mouth respiratory movements, it was not the same situation for inspiration and expiration movements. Although there was a light distinction between nasal inspiration and expiration, mouth inspiration and expiration distribution were almost overlapped, except for RMSRatio and roll-off features.

Regarding Mel Frequency Cepstral Coefficients, the coefficients distributions showed that, individually, some coefficients are most robust than others. For example, coefficient 2 clearly differentiates between nasal and mouth, and coefficients #4 and #11 do distinguish between mouth inspiration and expiration.

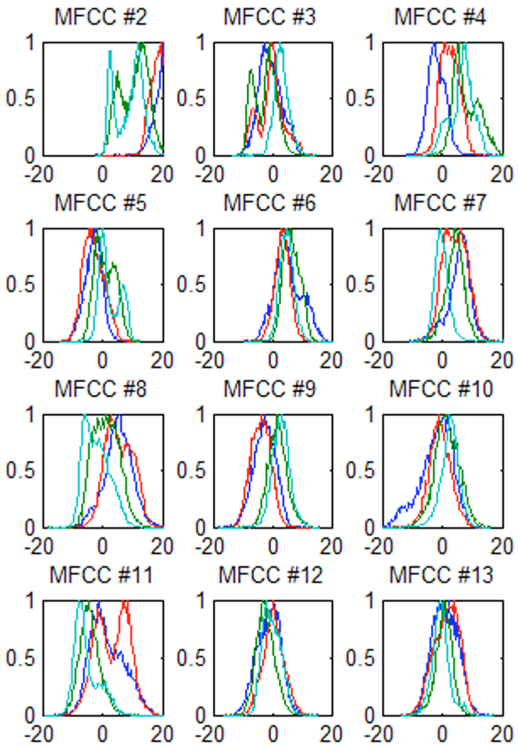

Fig. 4. Plot of MFCC 2 to 13 showing normalized distributions from the 4 respiratory movements analysed at 22,050 Hz of sampling rate and 92 ms of window size.

3.2. Training data set validation

In order to validate the training data set, cross validation method was applied. The data set was divided randomly into three parts. One third was used as samples to classify and the other two thirds were use as the model. This validation was done three times, rotating the roles of every third. After, the whole training data set was used to classify new unknown samples. The results are shown in Table 2.

Feature	Accuracy (%) Cross validation	Accuracy (%) Unknown Samples
RMS	25.8	26.1
SC	47.4	45.6
SF	47.9	44.6
RO	44.2	34.5
ZCR	42.1	31.8
RMSRatio	48.6	44.4
MFCC2	46.7	45.9
MFCC3	33.9	21.1
MFCC4	43.6	31.9
MFCC5	36.8	27.7
MFCC6	31.4	28.4
MFCC7	35.6	36.2
MFCC8	35.7	28.2
MFCC9	39.1	36.6
MFCC10	31.5	24.6
MFCC11	38.4	38.2
MFCC12	28.5	25.6
MFCC13	28.6	18.5
MFCC(2-13)	95.9	69.5
ALL	96.6	64.7

Table 2. Results from cross validation and from unknown samples. Done at 22,050 Hz of sampling and 92 ms window size.

The results showed that features used individually were not very robust. An accuracy of as much as 48% (with RMSRatio) when the probability of a random right classification was 25% means it is not a big success. However, taking multiple features simultaneously in the classification had high accuracy. Just all MFCC coefficients at the time gave an accuracy of 96% in cross validation and 69.5% for new unknown samples.

The same process was done for the rest of combinations of sampling rates and window sizes. The results were proportionally similar to those shown on Table 2. Nevertheless, changes in sampling rates and window sizes revealed some difference in the accuracy level, as shown in table 3.

The best results belonged to the combinations 22,050 Hz – 92 ms and 16,000 Hz – 128 ms, where accuracy was above 96.6%. Moreover, the combination 16 KHz – 128 ms were computationally more efficient: the samples per seconds decreased in 6,050 , the window was 36 ms wider and the samples in the training set were reduced from 48,685 to 35,362.

Sampling	Window length (ms)					
Hz	128	92-96	64-69	46	32	23
22.050		96.6	96.5	95.4		91.9
16.000	96.9	96.7	96.0		92.8	
11.025		96.5	95.8	94.0		87.9
8.000	96.2	96.0	94.6		89.1	

Table 3. Accuracy level using all simultaneous features.

Furthermore, the reduction of the sampling frequency from 16 KHz decreased accuracy, but very light and hardly perceptible (it continued above 96%).

The reduction of the window size, however, altered significantly the accuracy, whatever the sampling frequency was. Computationally, those results were desirable because wide window sizes and high sampling rates need much less data to process.

Furthermore, it was seen that features used individually had poor accuracy in comparison to the whole set of MFCC or all features. As the calculation of every feature had some CPU cost, cross validation accuracy was calculated for each combination from 1 to 17 simultaneous features (without considering RMS). That meant a total of 131,070 possible combinations and results are shown on figure 5 and table 4.

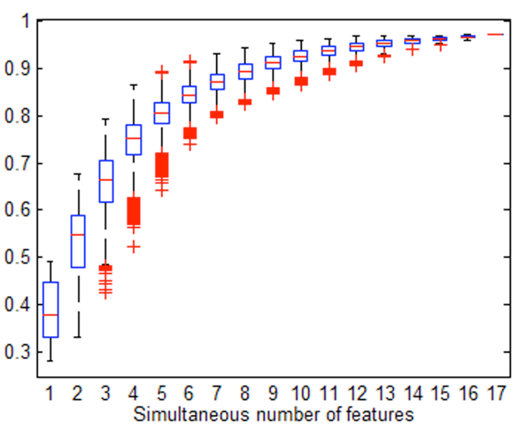

Fig. 5. Boxplot showing accuracy for every possible combination for every number of simultaneous features analysed at 22,050 Hz of sampling rate and 92 ms of window size.

#	Accuracy (%)	SC	SF	RO	ZCR	MFCC2	MFCC3	MFCC4	MFCC5	MFCC6	MFCC7	MFCC8	MFCC9	MFCC10	MFCC11	MFCC12	MFCC13	RATIO
1	49.0																	✓
2	67.5	✓																✓
3	79.1	✓					✓											✓
4	86.6			✓			✓	✓							✓			
5	89.4		✓				✓	✓		✓					✓			
6	91.5		✓				✓	✓	✓	✓					✓			
7	93.1					✓	✓	✓	✓	✓				✓	✓			
8	94.2						✓	✓		✓	✓			✓	✓	✓		✓
9	95.2						✓	✓	✓	✓	✓			✓	✓	✓		✓
10	95.9						✓	✓	✓	✓	✓			✓	✓	✓	✓	✓
11	96.3						✓	✓	✓	✓	✓	✓		✓	✓	✓	✓	✓
12	96.8						✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
13	96.8					✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
14	96.9				✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
15	96.9		✓		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
16	97.0		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
17	97.0	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

Table 4. Best combination for every simultaneous number of features.

Results from figure 5, showed that the average of accuracy increased when the number of features did it as well. But, it also showed a high difference in accuracy that exists for the same number of features. For example, four well-chosen features gave better accuracy than 10 bad selected; for five simultaneous features, the difference of accuracy between the best and worst combination is 25%. Table 4 shows the best found combination and accuracy level for each number of simultaneous features.

Simultaneous features	Best Comb. Accuracy (%) Cross validation	Unknown Samples
1	49.0	44.4
2	67.5	54.2
3	79.1	61.3
4	86.6	63.0
5	89.4	65.6
6	91.5	69.3
7	93.1	69.6
8	94.2	67.0
9	95.2	69.1
10	95.9	65.3
11	96.3	62.3
12	96.8	69.8
13	96.8	69.6
14	96.9	69.5
15	96.9	69.3

16	97.0	69.4
17	97.0	69.0

Table 5. Cross validation and unknown samples accuracy using best combination from Table 4.

Best combinations from Table 4 were used to run again cross-validation of training data set and classify new unknown samples. The results are shown in table 5, showing that accuracy grew quickly: with 6 simultaneous features, the accuracy was more than 90% in the cross validation and reached the highest value for unknown samples.

3.3. Computational cost

The solution had to identify respiratory movements in real time. So, breath sounds recording, sound segmentation, segment features extraction and labelling through the classifier have to be done at a faster pace than the data input speed, that is, the sampling rate. Otherwise, there would be buffer overruns that would imply loss of data.
Table 5 shows computational cost for two selections: one is the optimal combinations of sampling rate – window size and the other is the most CPU cost expensive one.

Function	Cost (ms)	
	22,050 Hz 92 ms	22,050 Hz 23 ms
Filtering	0.062	0.052
RMS	0.020	0.016
ZCR	0.062	0.036
Windowing & FFT	0.607	0.353
SC	0.661	0.401
SF	0.804	0.491
RO	0.842	0.531
12 MFCC	1.189	0.692
RMSRatio	1.257	0.785
kNN	7.495	27.978
Total	13.56	31.34

Table 6. Computational cost of iterative analysis functions involved in audio segment process and labelling.

As seen from the results, computational cost for configuration 22,050 Hz - 92 ms was far from its threshold (46 ms). So, keeping the window size high and lowering the sampling rate would increase that gap. However, computational cost for configuration 22,050 Hz - 23 ms overpassed its threshold (12 ms), due to the cost of the classifier, basically, as it had a number of records much higher. In this case, this configuration, even it has shown not to be as optimal, it would not run appropriately unless the software was executed in a faster computer.

3.4. Respiratory movement identification

Breath sound segments validation has been shown so far, using the training set as the classifier. The state machine implemented (section 2.5. The state machine) identified the respiratory movement as a whole.
A set of test were done, mixing mouth and nasal breathing from external subjects. The averages of positive identifications using the configuration of 22,050 Hz – 92 ms are displayed in Table 7.

Best combination of simultaneous features	Accuracy (%) Positive identifications
1	56.3
2	66.7
3	71.7
4	72.9
5	66.7
6	72.9
7	81.3
8	89.6
9	85.4
10	70.8
11	77.1
12	95.8
13	95.8
14	93.8
15	93.8
16	93.8
17	95.8

Table 7. Accuracy level of positive respiratory movements

Positive identifications were very high, even with a low number of features. Maximum accuracy was 95.8%, achieving this value with 12 simultaneous features.

3.5. User interface

The software runs in Matlab, either as a high level function or in a window interface. To run the solution in a GUI it is just necessary to call the *asbslab* module, which opens the window interface.

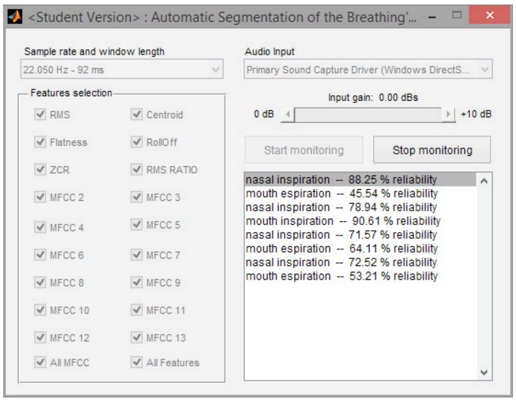

Fig. 6. User interface for the window version solution

The window interface version allows the user to choose the desired running configuration in a more friendly way, as shown in figure 6.
If users want to run the solution directly over the Mathlab console, it is just necessary to call the function *asbslabconsole* passing the following parameters:

- the combination of sampling frequency – window size as a string (from a predefined list).
- the set of chosen features to be used in the identification separated by the plus sign (+).

For example, executing the software using a sampling rate of 22,050 Hz, a window size of 92 ms and the features MFCC3, MFCC4 and RMSRatio, would be as following:

```
asbslabconsole('s22w92',ASBS_MFCC3+ASBS_MFCC4+
ASBS_RMSRATIO)
Monitoring now... Press CTRL+C to stop

mouth inspiration with 82.44% of reliability
mouth espiration with 70.80% of reliability
mouth inspiration with 85.39% of reliability
mouth espiration with 67.19% of reliability
nasal inspiration with 73.51% of reliability
nasal espiration with 89.80% of reliability
nasal inspiration with 70.44% of reliability
nasal espiration with 63.05% of reliability
```

Running the software using a sampling rate of 8,000 Hz, a window size of 128 ms and the features Centroid, MFCC3 and RMSRatio, would be as following (in this case the output did a false identification, marked here with two asterisks (**)):

```
asbslabconsole('s8w128',ASBS_CENTROID+ASBS_MFC
C3+ASBS_RMSRATIO)
Monitoring now... Press CTRL+C to stop.

nasal inspiration with 91.54% of reliability
mouth espiration with 75.35% of reliability
nasal inspiration with 95.00% of reliability
mouth espiration with 53.83% of reliability
**mouth espiration with 81.37% of reliability
mouth espiration with 68.43% of reliability
mouth inspiration with 88.96% of reliability
nasal espiration with 55.89% of reliability
nasal inspiration with 95.43% of reliability
nasal espiration with 57.25% of reliability
```

4 Conclusions

This paper has presented a software application to identify in real time the respiratory movements either mouth or nasal, which can be used for newer and practical monitoring solutions. The combination of signal processing analysis, classification method kNN and an implemented state machine has achieved more than 95% positive identifications. It has been implemented to work with predefined combinations of sampling frequencies and window segmentation sizes. Best accuracy is achieved at 22,050 KHz of sampling rate and 92 ms of window size. Also, it allows users to customize the subset of acoustic features implied in the respiratory movement identification. Moreover, the paper suggests, from each subset of simultaneous features, those that achieve higher accuracy. Computation cost has been proven to be far from critical, even for high sampling frequencies in combination with windows from middle sizes. This leads to think of the possibility to adapt the software to mobile phone platforms.

5 References

Chowdhury, S.K. & Majumder, A.K."Digital spectrum analysis of respiratory sound," *IEEE Trans. Biomed. Eng.*, vol. BME-28, pp. 784-788, 1981.

Charbonneau, G., Racineux, J.L., Sudraud, M. & Tuchias, E. "Digital processing techniques of breath sounds for objective assistance of asthma diagnosis," in *Proc. ICASSP,* Paris, France, 1982, pp. 736-738.

Petsatodis, T., Boukis, C., Maglogiannis, I. & Doukas, C. "Automated sleep breath disorders detection utilizing patient sound analysis," *Biomedical Signal Processing and Control*, pp. 256-264, 2012.

Moedomo, R.L., Mardiyanto, M.S., Ahmad, M., Alisjahbana, B. and Djatmiko, T."The breath sound analysis for diseases diagnosis and stress measurement," *System Engineering and Technology (ICSET), 2012 International Conference on*, pp. 1-6, 2012

Oud, M. and Dooijes, E.H. "Automated breath sound analysis," *Engineering in Medicine and Biology Society, 1996. Bridging Disciplines for Biomedicine. Proceedings of the 18th Annual International Conference of the IEEE,* vol. 3, pp. 990-992, 1996.

Brian, R. & Kain, A. S., "Automatic Classification of Breathing Sounds During Sleep," in *Proceedings of The 38th IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP).* Vancouver, Canada: IEEE, 2013.

Moussavi, Z. & Yadollahi, A. "Automatic breath and snore sounds classification from tracheal and ambient sound recordings," *Medical Engineering & Physics*, pp. 985-990, 2010.

Patel, U. "Computerized Respiratory Sound Analysis: An Exploration of Methods," Thesis, Department of Physics, CWRU School of Engineering, Cleveland, Ohio, 2011.

Avalur, D.S. "Human Breath Detection using a Microphone," M.S. Thesis, Faculty of mathematics and natural science, Univ. of Groningen, Groningen, NL, 2013.

Alshaer, H., Pandya, A., Bradley, T.D. and Rudzicz, F. "Subject independent identification of breath sounds components using multiple classifiers," *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*, pp. 3577 - 3581, 2014.

Yuan, K., Coyle, P. and Curran, D. "Using Acoustic Sensors to Discriminate between Nasal and Mouth Breathing," *International Journal of Bioinformatics Research and Applications*, vol. 7, no. 4, Sept-Dec 2011.

Mason, L."Signal Processing Methods for Non-Invasive Respiration Monitoring," Thesis, Dept. Eng. Sci., University of Oxford, Oxford, UK, 2002.

Young, S. J., Woodland, P.C., and Byrne, W. J. The HTK BOOK, Entropic Ltd., Jan. 1999.

Harrington, P. Machine Learning in Action. United States: Manning Publications Co., 2012.